REMARKS/ARGUMENTS

In response to the Advisory Action dated June 13, 2003, please consider the following remarks.

In the Advisory Action issued June 13, 2003, claims 1-15 remained rejected under 35 U.S.C. §102(e) as being anticipated by Dalal et al., U.S. Patent No. 6,064,999 (Dalal).

The applicant thanks the examiner for the courtesy extended during the July 18, 2003 telephonic interview. In that interview, the examiner indicated that Dalal did not re-process a SQL query to correct a prior SQL query.

Claims 1-15 are now pending in this application. Claims 1, 2, 7, and 9 have been amended to clarify the subject matter that the applicant considers to be the invention.

Each of the claims now pending in this application is believed to define an invention that is novel and unobvious over the prior art. Favorable reconsideration of this case is respectfully requested.

The present invention is not anticipated by, nor obvious in view of, the references relied upon in the Office Action issued March 6, 2003, as this prior art references do not disclose or suggest the claimed features of the present invention.

The Applicant respectfully submits that the present invention according to claims 1-15 is not anticipated by Dalal. Dalal discloses a method and system for efficiently performing database table aggregation. In a preferred embodiment, an

aggregation facility efficiently aggregates a source table using indices on an aggregated column of the source table and a grouping column of the source table. The facility uses the index on the aggregated column to identify the contents of the aggregated column in each row of the source table. The facility further uses information derived from the index on the grouping column to identify the contents of the grouping column in each row of the source table. For each row of the source table, the facility aggregates the identified aggregated column contents into a result value for the identified grouping column contents. In a further preferred embodiment, the facility generates a relation mapping from source table row to grouping column, which the facility uses to identify the contents of the grouping column in each row of the source table. In a further preferred embodiment, the facility may be used to perform multiple-level aggregations, as well as aggregations in which there are multiple grouping columns, multiple aggregated columns, and/or multiple result columns.

By contrast, claim 1 recites a query generator for generating a query for obtaining selected data from a database, the database having a number of detail tables in which data is stored, the query generator comprising a processor which is coupled to the database in use, the processor being adapted to: a. receive an input indicating the selected data to be obtained to generate a first query; b. analyse the input and determine whether the input requires a joining of data in a plurality of different detail tables, and an aggregation step; and, c. if so, causing the processor

to modify the input indicating the selected data to be obtained to generate a second query, the second query being adapted to cause the database to: i. aggregate the data within each of the plurality of detail tables as required; and, ii. join the aggregated data from each of the plurality of detail tables, the joined aggregated data representing the selected data.

Dalal does not disclose analysing the input and determining whether the input requires a joining of data in a plurality of different detail tables, and if so, causing the processor to modify the input indicating the selected data to be obtained to generate a second query being adapted to cause the database to aggregate the data within each of the plurality of detail tables as required and joining the aggregated data from each of the plurality of detail tables. Rather, Dalal discloses multi-level aggregation of data from a single source (detail) table. For example, Dalal discloses that:

> The facility also preferably permits the user to submit an aggregation query specifying a multiple-level aggregation--that is, specifying that aggregated values be formed for each of two or more grouping columns. For example:
>
> ```
> SELECT sum ([Sales Price]) FROM [Orders] LEVEL 2
> SELECT sum ([Sales Price]) FROM [Orders] LEVEL 1
> GROUP BY [Salesperson]
> GROUP BY [Division]
> ```

The above aggregation query specifies aggregating (that is, summing) the

Sales Price column of the Orders table, at level 1 for each Salesperson, and at level 2 for each division. (See col. 11, lines 51-64)

This example from Dalal clearly discloses aggregating data from only a single source (detail) table, the Orders table. By contrast, the present invention requires aggregating the data within each of the plurality of detail tables as required.

In addition, Dalal does not disclose joining the aggregated data from each of the plurality of detail tables. Rather, Dalal discloses performing multi-level aggregation on a single source (detail) table. In the multi-level aggregation disclosed by Dalal, a level 1 aggregation table is generated for each value of a particular column in the source table. As disclosed by Dalal, aggregation involves generating a value such as a sum. The level 1 tables are then aggregated to form a level 2 table by generating aggregated values, such as sums, from the values in the level 1 tables. (See col. 9, line 65 to col. 10, line 21) The level 1 tables disclosed by Dalal are not the source (detail) tables, but rather are aggregation tables generated from a single source table.

The level 1 tables are not joined as required by the present invention, but are merely aggregated again to form a higher-level aggregation table. As is well known in the database management art, a join is an operation in which every specified row in one table is matched with and merged into every specified row in

another table. By contrast, Dalal discloses aggregating the data in the level 1 tables to form a level 2 table, no join operation is disclosed or suggested.

Dalal does not disclose causing the processor to modify the input indicating the selected data to be obtained to generate a second query, the second query being adapted to cause the database to perform the recited aggregation and join steps. Dalal discloses receiving a query indicating data to be obtained and performing the received query. But Dalal does not disclose modifying the received query, nor does Dalal disclose modifying the received query to perform the required aggregation and join steps where the aggregation in the aggregation step was not previously contemplated.

Similarly, claim 9 recites a database system, the database system comprising: a. a database, the database comprising: i. a store for storing data, the store having a number of detail tables; and, ii. a database processor coupled to the store for obtaining data in accordance with a received query; and, b. a query generator for generating a structured query for obtaining selected data from the database, the query generator comprising a processor adapted to: i. receive an input representing a first structure query to be generated; ii. analyse the input and determine whether the input requires a joining of data in a plurality of different detail tables, and an aggregation step; and, iii. if so, causing the processor to modify the input indicating the selected data to be obtained to generate a second structured query, wherein the database processor responds to the second structured

query to: (1) aggregate the data within each of the plurality of detail tables as required; and, (2) join the aggregated data from each of the plurality of detail tables, the joined aggregated data representing the selected data.

As in claim 1, Dalal does not disclose analysing the input and determining whether the input requires a joining of data in a plurality of different detail tables and aggregating the data within each of the plurality of detail tables as required and joining the aggregated data from each of the plurality of detail tables; joining the aggregated data from each of the plurality of detail tables; or causing the processor to modify the input indicating the selected data to be obtained to generate a second structured query, the second structured query being adapted to cause the database to perform the recited aggregation and join steps.

Thus, the present invention, according to claims 1 and 9, is not anticipated by Dalal. Likewise, the present invention, according to claims 2-8 and 10-15, which depend from claims 1 or 9, is not anticipated by Dalal.

In view of the above, it is respectfully submitted that the present invention is allowable over the references relied upon in the Office Action. Accordingly, entry of the amendment and favorable reconsideration of this case and early issuance of the Notice of Allowance are respectfully requested.

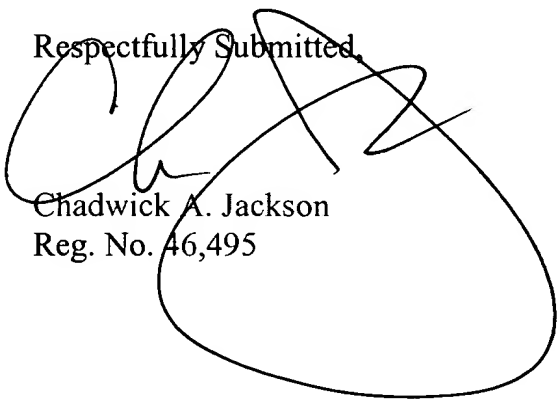
Additional Fees:

The Commissioner is hereby authorized to charge any insufficient fees or credit any overpayment associated with this application to Deposit Account No. 19-5127 (19111.0045).

Conclusion

In view of the foregoing, all of the Examiner's rejections to the claims are believed to be overcome. The Applicants respectfully request entry of the amendment and reconsideration and issuance of a Notice of Allowance for all the claims remaining in the application. Should the Examiner feel further communication would facilitate prosecution, he is urged to call the undersigned at the phone number provided below.

Respectfully Submitted,



Chadwick A. Jackson
Reg. No. 46,495

Dated: August 6, 2003

Swidler Berlin Shereff Friedman, LLP
3000 K Street, N.W., Suite 300
Washington, D.C. 20007
(202) 424-7500